■1 群（信号・システム）- 3 編（暗号理論）

14 章 サイドチャネル攻撃と耐タンパー技術

Side-channel attacks and anti-tamper techniques

（執筆者：崎山一男）［2019 年 1 月 受領］

■概要■

暗号技術をシステムで利用する際には，暗号アルゴリズムを専用ハードウェアや CPU 上のソフトウェアに実装する．つまり，暗号技術やその応用技術は，半導体集積回路によって実現されている．集積回路を動作させる際に必要となる電気エネルギーは，処理中の演算やデータに依存する．そのため，回路内部の電流の変化に伴い生じる物理現象の時間変化を観測することで，回路中で処理されている演算やデータに関する情報を得ることができる．ここでは，消費電力や漏洩電磁波を物理情報と呼び，暗号文を公開情報として扱う．

暗号アルゴリズムをハードウェアやソフトウェアに実装した場合，物理情報が攻撃者にとって最も貴重な情報となりうる．物理情報が漏洩する経路はサイドチャネルと呼ばれ，サイドチャネルを用いて暗号アルゴリズムの秘密情報を読み出す攻撃は，サイドチャネル攻撃と呼ばれる．サイドチャネル攻撃における攻撃者の主な目的は，公開鍵暗号のプライベート鍵や共通鍵暗号における秘密鍵の特定である．

本章では，代表的なサイドチャネル攻撃を例示し，その対策技術について述べる．また，物理情報を受動的に観測するのではなく，回路内部の情報を意図的に操作するアクティブ型の攻撃であるフォールト攻撃を併せて概説し，サイドチャネル攻撃との関係を示す．

【本章の構成】

14-1 章では暗号実装への攻撃における分類を概説し，サイドチャネル攻撃の位置付けを概説する．14-2 章では差分電力解析／差分電磁波解析を紹介する．14-3 章では差分故障解析を説明し，14-4 章では耐タンパー技術について紹介する．

■1 群 - 3 編 - 14 章

14-1 暗号実装への攻撃

（執筆者：崎山一男）[2019 年 1 月 受領]

ここでは，暗号アルゴリズムが実装されたソフトウェアやハードウェアから秘密情報を取得する攻撃の概要について説明する．

14-1-1 攻撃の分類

図 14･1 は分類の一例である．暗号実装への攻撃は，暗号アルゴリズムの入出力データを用いた暗号解析と呼ばれる理論的な解析と，物理情報を併用する物理攻撃とに大別できる．暗号解析において攻撃者は，攻撃に必要となる膨大な計算量を処理するか，暗号アルゴリズムの瑕疵を見つけ攻撃に必要となる計算量を削減するかのいずれかを目指す．物理攻撃では，物理情報を用いて後者の実現を狙う．

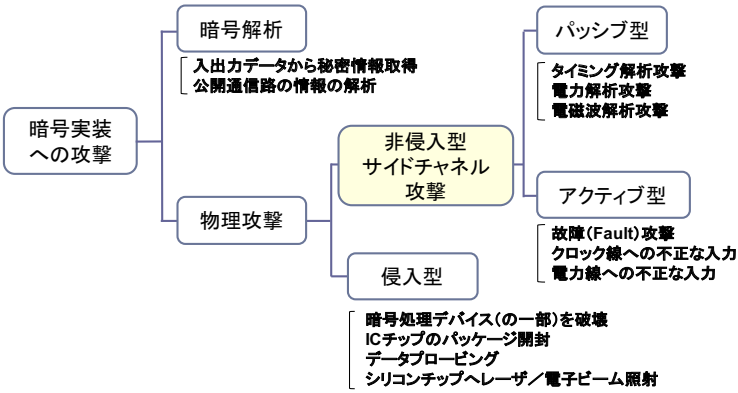

図 14･1　暗号実装への攻撃の分類

（この例ではアクティブ型のサイドチャネル攻撃をフォールト攻撃とする．）

物理攻撃は，大きく侵入型の攻撃と非侵入型の攻撃に分けることができる．前者は，物理情報を効率よく得るために，IC チップのパッケージを開封したり，回路の一部を破壊したりする攻撃である．そのため，攻撃のコストは高くなるとともに，攻撃の痕跡を残すことになる．後者の非侵入型の物理攻撃をサイドチャネル攻撃という．攻撃に必要となる装置は侵入型攻撃と比べて安価なものでよいが，秘密情報の導出の計算コストは高くなる．しかし，十分な物理情報を得られれば，計算量は大幅に削減でき，汎用の計算機でも秘密情報を特定できる．

14-1-2 サイドチャネル攻撃

サイドチャネル攻撃は，さらにパッシブ型とアクティブ型に分けられる．パッシブ型においては，いくつかの解析手法があり，タイミング解析，電力解析，電磁波解析が有名である．これらの攻撃において，攻撃者は，演算処理の時間差，演算処理中の消費電力，漏洩電磁波を利用する．アクティブ型はフォールト攻撃といい，サイドチャネル攻撃と同列に分類する

こともある．クロック端子や電力端子から，異常な信号を物理的に入力し，暗号アルゴリズムに誤動作を誘発することで，内部信号に関する情報を得る．

サイドチャネル攻撃の先駆けは，公開鍵暗号である RSA 暗号に対するタイミング解析[2)]である．RSA 暗号の復号における処理タイミングの違いから，ビットごとにプライベート鍵を入手するものである[3)]．さらに共通鍵暗号に対するサイドチャネル攻撃として，差分電力解析が提案されている[4)]．詳細については，後に述べる．

14-1-3 フォールト攻撃

一方，フォールト攻撃は，CRT（Chinese Remainder Theorem）を用いた公開鍵暗号 RSA に対するものが最初である[5)]．いわゆる Bellcore 攻撃として知られており，暗号アルゴリズムの出力に影響を与え，正しい出力値とエラー出力値を比較・解析する差分故障解析を最初に行ったものである．その後，攻撃の対象は共通鍵暗号へと拡張された[6)]．近年，発表が多く見られる AES 暗号へのフォールト攻撃の多くはこの手法にもとづく．他にも，故障が暗号アルゴリズムの出力に影響しないことを利用したセーフ・エラー攻撃[7)]や，故障の発生の有無から内部信号を読み取るインエフェクティブ攻撃が知られている[8)]．なお，故障誘発の手法として，IC チップに直接レーザーを照射したり，回路の配線からデータをプローブする場合は，侵入型の攻撃に分類される．

■1 群 - 3 編 - 14 章

14-2 差分電力解析／差分電磁波解析

（執筆者：崎山一男）[2019 年 1 月 受領]

サイドチャネル攻撃の基本は，暗号アルゴリズム処理における物理情報に対する受動的観測にある．サイドチャネル攻撃を目的として収集した物理情報を，サイドチャネル情報という．サイドチャネル情報を解析することにより，ブラックボックスとして仮定していた暗号アルゴリズムの中間データを入手できるため，暗号解析に必要となる計算量を大幅に削減することができる．以降，AES ブロック暗号の暗号化処理を例にしてサイドチャネル解析を説明する．

AES 暗号化処理における出力は，公開情報として扱える暗号文である．サイドチャネル攻撃では，暗号化処理中の消費電力や漏洩電磁波をサイドチャネル情報として取得する．もし，サイドチャネル情報から暗号アルゴリズムの中間値を導出することができれば，中間値と暗号文から秘密鍵を導出することができる．しかし，この方法には大きな問題がある．それは，測定ノイズやアルゴリズミックノイズといった攻撃対象に関する信号以外のノイズの影響である．つまり，SN 比が極めて低いサイドチャネル情報から，直接中間値を導出できないのである．

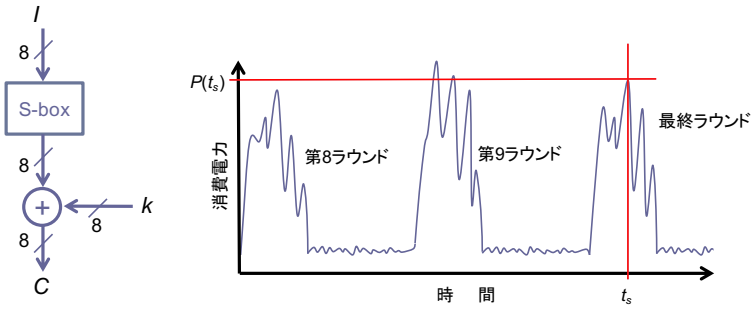

図 14･2　AES ブロック暗号回路の暗号化処理における電力消費のイメージ図．

そこで，鍵を予測したうえで，サイドチャネル情報をモデル化し，観測した複数のサイドチャネル情報との相関を調べる手法が考案された．図 14･2 に示すように，AES 暗号化処理の最終ラウンドにおける中間値 I は，非線形演算である S-box（S）で処理された後に，ラウンド鍵 k と XOR 演算され，暗号文 C として出力される．中間値 I は，予測鍵 k_g を用いて $I = S^{-1}(C \oplus k_g)$ と計算できる．モデル化の対象とする中間値は，I のように公開情報（C）を非線形演算で処理したものが良い．非線形演算の性質のおかげで，誤った予測鍵で導出した I と実際のサイドチャネル情報との相関が低くなるためである．

I に対応するサイドチャネル情報のモデルには，モデル精度が高く，汎用性もあるハミング距離モデルがよく使われる．ハミング距離モデルは，連続した 2 ラウンドにおける中間値の変化をサイドチャネル情報モデルとするものである．例えば，1 サイクルで 1 ラウンドの処理を行う AES 暗号化処理ハードウェアの場合，i ラウンドと $i+1$ ラウンドにおける中間

値のハミング距離がハミング距離モデルとなる $(i = 0,1,2,\ldots,9)$．l 個の異なる平文に対して，AES 暗号化処理を実行した際のサイドチャネル情報のモデルを，以下のように表すことにする．

$$\mathrm{W}_l^{k_g}[i] = \left(W_1^{k_g}[i], W_2^{k_g}[i], \ldots, W_l^{k_g}[i]\right). \tag{14･1}$$

ここで k_g は予測した秘密鍵である．ベクトルの各要素は，暗号文と予測した秘密鍵から求めた，i ラウンドと $i+1$ ラウンドにおける中間値のハミング距離である．

一方，取得したサイドチャネル情報は，以下のように整理することができる．

$$\mathrm{P}_l^k(t_s) = \left(P_1^k(t_s), P_2^k(t_s), \ldots, P_l^k(t_s)\right). \tag{14･2}$$

ここで k は暗号化処理に用いられた秘密鍵である．ベクトルの各要素は，時刻 t_s における消費電力である．最後に，すべての秘密鍵を予測したモデルと測定したサイドチャネル情報との相関を導出する．

$$\rho\left(\mathrm{W}_l^{k_g}[i], \mathrm{P}_l^k(t)\right). \tag{14･3}$$

特定のバイト演算に対して攻撃を行う場合，攻撃で予測する鍵の数は 256 個となる．攻撃者の計算能力の許す限り，鍵の候補数を増やすことができる．時刻 t_s で，所望のサイドチャネル情報が漏洩していれば，$k_g = k$ で相関が大きくなるはずである．もし，どの予測鍵でも高い相関が得られなかった場合には，t_s を変化させるか波形数 l を増やす．

■1 群 - 3 編 - 14 章

14-3 差分故障解析

（執筆者：崎山一男）[2019 年 1 月 受領]

ここでは，AES ブロック暗号に対する効率の良い差分故障解析としてピレとキスカテールの攻撃[9]を紹介する．

14-3-1 AES ブロック暗号に対する差分故障解析

この攻撃における差分の伝搬の様子を図 14･3 に示す．4×4 のマス目は 128 ビットの中間値のステートを表し，1 つのマス目が 1 バイトに相当する．攻撃者により誘発された 8 ラウンド目の故障は，AES 暗号のアルゴリズムに従って拡散し，出力である暗号文のすべてのバイトに影響を与える．

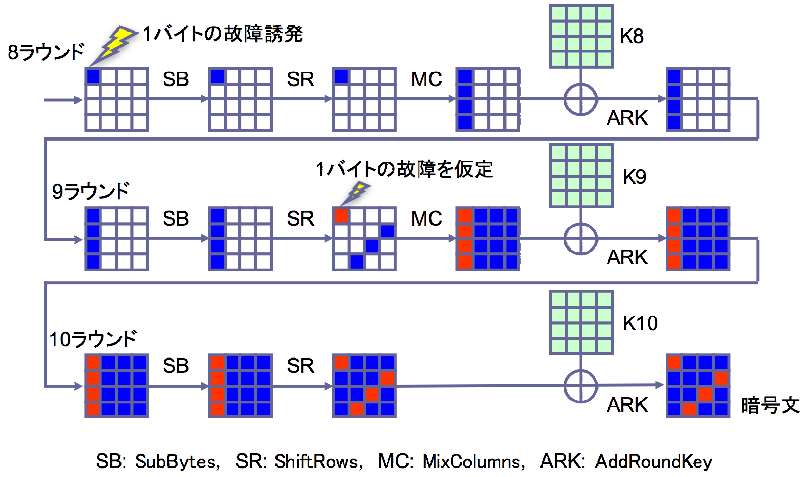

図 14･3　AES ブロック暗号回路への差分故障解析における差分拡散の様子．

いくつかある解析手法の中で，暗号文の差分からフォールトが発生した 1 バイトフォールトまで遡って計算をするのが最も素直である．ShiftRows 処理，MixColumns 処理，及び AddRoundKey 処理については，出力差分から入力差分を計算する．非線形演算である SubBytes 処理については，出力差分から入力差分を計算することができないため，鍵を予測して入力差分を求める．このようにして，8 ラウンド目の入力における差分が 1 バイトとなるような鍵を導出すればよい．

しかしこのままでは，予測する鍵の組合せが膨大となるため，9 ラウンド目に 1 バイトの差分が 4 回誘発されたものとし，解析を 4 つに分割する．こうすることで，それぞれの解析で予測する 10 ラウンド目の鍵は，2^{32} 個である．9 ラウンド目の MixColumns 処理の入力差分が 1 バイト分しかないことに着目し，予測した 2^{32} 個の鍵は約 2^8 個に削減できる．4 分割した解析により，10 ラウンド目の鍵の組合せは $(2^8)^4 = 2^{32}$ 通りとなる．最後に，8 ラウ

ンド目の MixColumns 処理の入力差分が 1 バイト分しかないことに着目すれば，10 ラウンド目の鍵を約 2^8 個に削減することができる．

14-3-2 攻撃効率

ピレとキスカテールの攻撃で鍵を特定するには，1 回の解析で残った約 2^8 個の鍵候補を総当たりで調べる．ただし，攻撃者が平文と暗号文のペアを入手していることが条件である．もし，平文と暗号文のペアがない場合でも，差分故障解析をもう一度繰り返せば高い確率で鍵を特定できる．このように，ほんの数回の攻撃試行で秘密鍵のすべてのビットを知ることができる．差分故障解析は，故障誘発が可能であれば非常に効率の良い攻撃といえる．

■1 群 - 3 編 - 14 章

14-4 耐タンパー技術

（執筆者：崎山一男）[2019 年 1 月 受領]

ここでは，サイドチャネル攻撃に対する基本的な対抗技術を紹介する．つまり，本節で述べる耐タンパー技術は，暗号実装から秘密情報が読み出されないような実装上の対策技術を指す．主なものは，サイドチャネル攻撃に対するアルゴリズムでの対策や回路上の工夫によるものである．

14-4-1 ハイディング対策

ディジタル回路におけるハイディング技術は，相補的な動作により実現できる．例えば AND 演算の場合には，以下のように正論理と負論理を同時に演算する．

$$\begin{cases} z = x \wedge y \\ \neg z = \neg x \vee \neg y. \end{cases} \tag{14·4}$$

これにより，すべての入出力信号のハミング重みは常に 1 となるため，消費電力から内部の信号が類推できなくなることが期待できる．なおハイディングにおいて，乱数生成器を必要としないことは，実装上の大きな利点である．また，回路を空間的に 2 重化しているため，フォールト攻撃に対する一定の耐性を与えられることも魅力である．しかしながら，ゲート間の配線容量のわずかな違いなどから，信号遷移の時間や消費電力を完全に等しくすることは難しい．そのため，次に述べるマスキング対策と比べて，サイドチャネル攻撃への耐性向上は限定的との見方もある．

14-4-2 マスキング対策

マスキング対策の基本は，内部信号をランダムにし，サイドチャネル攻撃を困難にすることにある．そのため乱数が必要となる．以下は，n ビットのデータ a を n ビットの乱数 r でマスキングする例である．

$$a_r = a \oplus r. \tag{14·5}$$

これをブーリアンマスキングという．演算中の乱数の鮮度を維持するために，乱数を適切に変更するリマスキングを行う．また，最後に乱数を取り除くアンマスキングを行う．線形演算とブーリアンマスキングの親和性は高く，実装が容易である．しかし，非線形演算に対しては工夫が必要となる．

マスキングの別の見方として，スレッショルド実装が提案されている[10]．a，b を乱数によりシェアと呼ばれる変数に分割し，それぞれ (a_1, a_2, a_3)，(b_1, b_2, b_3) と表す．ただし，$a = a_1 \oplus a_2 \oplus a_3$，$b = b_1 \oplus b_2 \oplus b_3$ である．以下は，3 分割したデータに対する AND 演算処理 $c = a \wedge b$ の例である．

$$\begin{cases} c_1 = (a_2 \wedge b_2) \oplus (a_2 \wedge b_3) \oplus (a_3 \wedge b_2) \\ c_2 = (a_3 \wedge b_3) \oplus (a_1 \wedge b_3) \oplus (a_3 \wedge b_1) \\ c_3 = (a_1 \wedge b_1) \oplus (a_1 \wedge b_2) \oplus (a_2 \wedge b_1). \end{cases} \tag{14·6}$$

ここで，$c = c_1 \oplus c_2 \oplus c_3$ である．分割したそれぞれの AND 演算にはすべてのシェアを用いていないため，サイドチャネル攻撃に対する一定の耐性を有する．ただし，実装した回路からのサイドチャネル情報は，上記の式で表現されないものを含むため，IC チップを用いた安全性評価が重要である．

■参考文献

1) 崎山 一男, 菅原 健, 李 陽, “暗号ハードウェアのセキュリティ,” コロナ社, 2019.
2) P.C. Kocher, “Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS and Other Systems,” *Advances in Cryptology – CRYPTO ’ 96*, LNCS1109, Springer-Verlag, pp.104–113, 1996.
3) 崎山 一男, 太田 和夫, “暗号への脅威「サイドチャネル攻撃」とその対策,” 「科学」報告・解説, 岩波書店, Vol.78, No.10, pp.1080-1083, 2008.
4) P.C. Kocher, J. Joshua, and B. Jun, “Differential Power Analysis,” *Advances in Cryptology – CRYPTO ’ 99*, LNCS 1109, Springer-Verlag, pp.388–397, 1996.
5) D. Boneh, R. DeMillo, and R. Lipton, “On the Importance of Checking Cryptographic Protocols for Faults (Extended Abstract),” *Proc. EUROCRYPT’97*, pp.37–51, 1997.
6) E. Biham and A. Shamir, “Differential Fault Analysis of Secret Key Cryptosystems,” *Advances in Cryptology – CRYPTO ’ 97*, LNCS 1294, Springer-Verlag, pp.513–525, 1997.
7) S. M. Yen, and M. Joye, “Checking Before Output May Not Be Enough Against Fault-Based Cryptanalysis” *IEEE Transactions on Computers*, vol.49, no.9, pp.967–970, 2000.
8) T. Sugawara, N. Shoji, K. Sakiyama, K. Matsuda, N. Miura, and M. Nagata, “Exploiting Bitflip Detector for Non-Invasive Probing and its Application to Ineffective Fault Analysis,” *Proc. FDTC ’ 17*, IEEE, pp.49–56, 2017.
9) G. Piret, J.-J. Quisquater, “A Differential Fault Attack Technique against SPN Structures, with Application to the AES and KHAZAD,” *Proc. CHES’03*, LNCS 2779, Springer-Verlag, pp.77–88, 2003.
10) S. Nikova, C. Rechberger, V. Rijmen, “Threshold Implementations Against Side-Channel Attacks and Glitches,” *Proc. ICICS’06*, LNCS 4307, Springer-Verlag, pp.529–545, 2006.